

# SECURING RESOURCES ON WINDOWS 2000 SERVERS

**After reading this chapter and completing the exercises  
you will be able to:**

- ◆ Understand the various authentication protocols used in Windows 2000.
- ◆ Understand how access tokens and access control lists help manage authorization to network resources.
- ◆ Secure file resources using NTFS and share permissions.
- ◆ Secure file resources using the Encrypting File System.
- ◆ Secure printer resources.
- ◆ Secure the Windows 2000 registry.
- ◆ Plan and configure a security audit policy.

One of the first components of an effective Windows 2000 security design is controlled access to the various resources available on the network. Most companies now store almost all corporate data on network servers. In many companies, that data may be the corporation's most valuable asset, as it may include all of the customer billing information, the design documents for any manufacturing process, and the corporate financial documents. This data must be accessible to the people who need access, and the people who do not require access must be prevented from accessing the information.

Protecting the network resources begins with an understanding of the various authentication and access methods available in Windows 2000 and the impact that down-level clients have on the authentication plan. It is also important to have a solid understanding of how to design and implement secure resource access, including such concepts as NTFS and share permissions, the encrypting file system, printer security, and a secure registry.

This chapter illustrates the concepts, features, and procedures used to control who logs onto the Windows 2000 network, and who gains access to the network services and resources. The steps to plan and configure an audit policy are also discussed to ensure that the security goals are accomplished and maintained.

The first step in securing network resources is to establish **user authentication** before granting access to a resource. This prevents users that are unknown to the system from accessing any available network resources.

User authentication consists of two main processes. The first process is called an **interactive login**. During this process the user logs onto, and is authenticated by, either a local computer or a domain controller. The user is authenticated and granted access to the network system, but may not necessarily have access to all resources. The second process is called **network authentication**. This is the process of identifying the user to each specific network service as needed, and allowing or denying access based upon permissions applied for access to the service.

The process of network authentication is similar to many situations we deal with every day. For example, if you want to rent a car or access a secure area in a building, you usually have to show some form of identification, such as a driver's license or a security badge. Access to the rental car, or the secure area is granted based upon possession of a valid card, with additional security in the form of a photograph (photo id). In a networking situation, authentication is treated in a similar manner. Authentication is granted based on user identification in combination with some form of shared "secret." This **shared secret** can be in the form of a password, a PIN number, or a special key that is only known to the person requesting access and the server that is authenticating the user. Various authentication protocols have been developed to assist in sharing and protecting this secret.

In addition to the shared secret models for authentication, Windows 2000 also supports other forms of authentication. One example is hardware-based authentication using smart cards. A **smart card**, which is similar to an ATM bank card, stores user authentication information, such as a certificate and private keys. Because the Windows 2000 authentication methods are extendable, it is also possible to use **biometric systems**, such as fingerprint or retinal scanning, which are starting to become popular in very secure environments.

---

## IMPLEMENTING USER AUTHENTICATION

Before a user can access any resource on the network, the user must be authenticated. The process of authentication confirms the user's identity. Once the user's identity has been confirmed, the user account is then the basis for controlling access to resources on the network for that user.

The process of authentication begins when a user enters an account name and password (or shared secret) at the logon screen. If these credentials correspond to information previously placed in the directory database of the authenticating local computer or domain controller, then the user is given access to the network. As stated earlier, the shared secret could be in the form of a password, a PIN number, or images of a scanned fingerprint or retina. To assist in keeping this secret secure, numerous authentication protocols have been developed over the years. The use of a particular protocol depends on what service is being requested, which operating system is being used, and the security needs of a company. The next section of this chapter will discuss authentication protocols such as Kerberos v5, NTLM, and certificate-based authentication.

## Kerberos version 5 Authentication

Kerberos version 5 authentication is the default protocol for network authentication for all Windows 2000 computers. This protocol allows users to provide a single account name and password combination to access services and resources throughout the Windows 2000 network. Kerberos version 5 is based on RFC 1510, and will interoperate with other servers and clients that follow the RFC specifications.



Windows 2000 client computers connecting to Windows 2000 domain controllers can use Kerberos on a Microsoft network. The issues of supporting earlier or down-level clients will be addressed later.

Kerberos is used both for network authentication and for authorization to access network resources. These processes for gaining access to network resources include:

- The Windows 2000 client who is requesting access
- The Key Distribution Center (KDC), which is the Windows 2000 domain controller
- The Windows 2000 server with the resource or service that is being requested

In the Windows 2000 implementation of Kerberos, the **Key Distribution Center (KDC)** is typically the domain controller that stores the directory database containing all users and passwords. The KDC provides two main services for the security of the network:

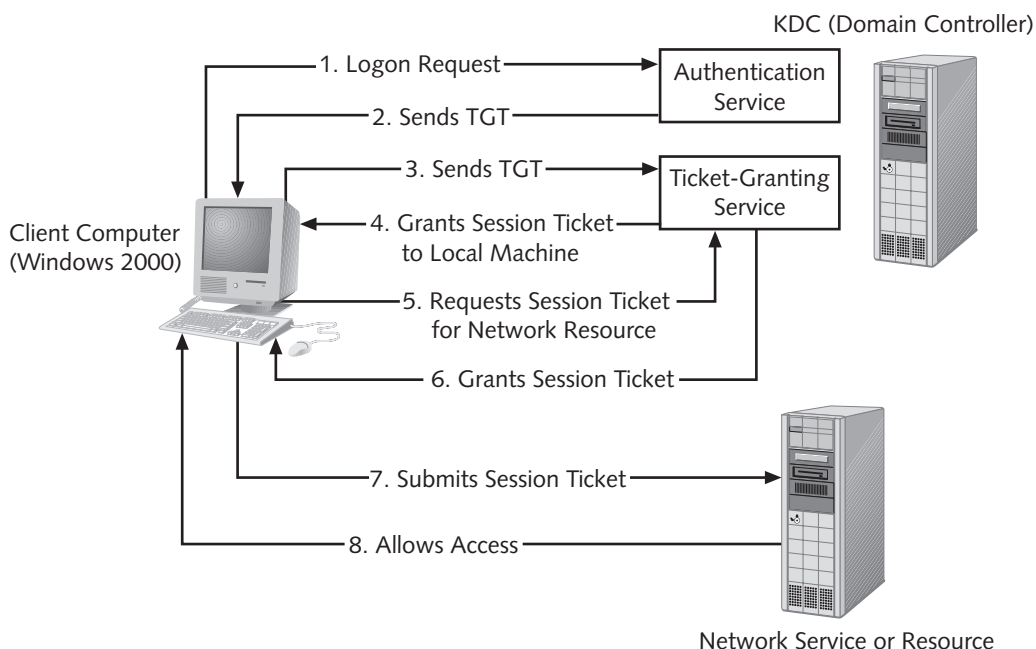
- **Authentication Service**—Authenticates and issues Ticket Granting Tickets to users.
- **Ticket-Granting Service**—Issues Session Tickets for access to network resources.

The process of authentication and granting access to resources requires two tickets from the KDC. The first ticket is the **Ticket Granting Ticket (TGT)** and is issued to the user during a successful logon. The TGT simply gives the user access to request session tickets. By default, the TGT has a lifetime of 10 hours, which means that the same ticket can be used to acquire session tickets for the 10-hour period without going back to the KDC for

a new TGT. The default lifetime of the TGT can be adjusted in the security policy of the domain. Each time that a user needs to access a resource, the TGT is submitted to the Ticket-Granting Service on the KDC. The KDC determines whether the user has the permission needed to access the resources, and if access to the resource is granted, a **session ticket** is sent back to the user's machine. The client machine stores the ticket in the client ticket cache system. Each time that the user needs access to the resource or service, the client presents the cached session ticket to the target resource. The session ticket has a default lifetime of 600 minutes (10 hours), and can also be adjusted in the security policy of the domain.

## Kerberos in Action

The process of Kerberos authentication and granting access to network resources is fairly complicated. Figure 3-1 illustrates the process.



**Figure 3-1** Kerberos authentication process

**Step 1** When a user logs on by typing a username and password, the client computer sends a packet to the KDC. The packet includes:

- The user name
- A secret key, derived from a hash of the user's password
- A time stamp
- A request for a Ticket Granting Ticket (TGT)

The entire packet, except the user name, is encrypted using the secret key. When the packet arrives at the server, the server looks at the username and then checks the directory database for the secret key (password) associated with the user's account. The server then decrypts the encrypted data in the packet using the user's secret key and checks the time stamp. If the decryption is successful, and the time stamp is within five minutes of the current time on the server, the server authenticates the user. If the decryption failed, then the user must have entered the wrong password, and the authentication fails. If the time stamp is more than five minutes old, the authentication will also fail. By default, Windows 2000 requires a time difference of less than five minutes between the workstation and the domain controller. The reason for the small allowable time difference is to prevent someone from capturing the authentication packets and then replaying them at a different time. The maximum allowable time difference can be configured on the domain security policies.

**Step 2** After the user is authenticated, the server sends the user a TGT. The packet is also time-stamped and encrypted using the secret key. When the packet arrives at the client computer, the user's secret key is used to decrypt the packet. If the decryption is successful, and the time stamp is valid, then the user's computer assumes that the KDC is authentic because it knew the user's secret key. The TGT is then cached on the local machine for 10 hours.

**Steps 3 and 4** After the user has been authenticated, the user must acquire a session ticket in order to log on to the client computer. A request is sent to the KDC for a session ticket for the local machine. The client presents the KDC with the TGT, and if it is valid, the client computer will receive the session ticket. This ticket is used to help build the access token and local rights for the user who has logged on.

**Step 5** When the client needs access to a network resource, such as a file on a file server or a mailbox on an Exchange server, the request is sent to the KDC for another session ticket. The request includes the TGT, a time stamp, and a shared secret, and it is encrypted using the session key that was acquired during logon.

**Step 6** The server decrypts and checks the data in the packet. If the data is acceptable, the server will issue the session ticket to the client.

**Step 7** The client now presents that session ticket to the network service to gain access.

**Step 8** The client now has access to the server resources. If the client needs subsequent use of the resource or service, the session ticket is pulled up from the ticket cache and reissued to the target resource server. If the session ticket has expired, the client has to return to the KDC to obtain a new ticket.



This process of obtaining a session ticket from the KDC before accessing a network resource is completely different from the process used in Windows NT. When a client tried to access a resource on a Windows NT server, the client would connect directly to the resource and request access. The server holding the resource would then use a process of pass-through authentication and connect to the domain controller to check whether the user had the right level of permission. With Kerberos, the client does not connect to the resource until it has received a session ticket from the KDC.

## NTLM Authentication

**NTLM authentication (Windows NT LanManager)** is the second option when authenticating a user on a network and is supported mainly for backwards compatibility with Windows NT 4.0 and Windows 9x client computers. This protocol is used in various scenarios, as follows:

- A Windows 95/98/NT-based computer authenticates to a Windows 2000 Domain Controller. The Directory Services Client must be installed for Windows 95/98 computers; otherwise these operating systems can only authenticate using the LAN Manager protocol.
- A Windows 2000 computer authenticates to a Windows NT-based server.
- A logon request is sent to a Windows 2000 standalone server.
- A fallback is necessary in case a Windows 2000 client trying to log on to a Windows 2000 domain controller is unable to authenticate by using the Kerberos protocol.
- Authentication to a Windows 2000 Cluster Server environment is required.

The NTLM protocol is significantly less secure than Kerberos, and many password-cracking tools have been developed to decrypt NTLM authentication. With Windows NT 4.0 Service Pack 4, Microsoft introduced a new version called NTLMv2. This new version includes additional security, such as unique session keys each time a new connection is established, and an advanced key exchange to protect the session keys. The main difference between Kerberos and NTLM is in the way that a server authenticates a client's attempt to access a resource. As previously discussed, when a client wants to access a resource using Kerberos, a session ticket is presented to the target resource server directly from the client. NTLM requires that the target resource server contact the domain controller to validate the user's credentials.

As a result of the security concerns with NTLM, your security plan should eliminate the use of NTLM wherever possible. All Windows 2000 clients should be required to use Kerberos. This is the default configuration for Windows 2000, but if the Windows 2000 client cannot use Kerberos (for example, if the Kerberos authentication ports are blocked on a router) then the Windows 2000 client will fall back to using NTLM. As you test your security plan, ensure that all Windows 2000 clients are using Kerberos. If they aren't, you need to determine what is preventing them from doing so.

If your network requires a high level of security, your security plan should include a recommendation to upgrade all clients to Windows 2000 to ensure that only Kerberos is used for authentication.

## Down-level Client Authentication

3

Almost all large corporate networks have a mixture of client operating systems in use, including Windows 95, Windows 98, and Windows NT 4.0. All of these clients create a security concern when implemented within a Windows 2000 network. In particular, Windows 95/98 clients are the most insecure because they use the LAN Manager (LM) authentication, which is one of the weakest authentication protocols available. Windows NT 4.0 used NTLM, which can be updated to version 2 in Service Pack 4.

To help remedy these security concerns, the Directory Services Client is available. This add-on component to Windows 95/98 enables these clients to use NTLM version 2 on the Windows 2000 network. The Directory Services Client, which is also available for Windows NT, implements additional features, such as Active Directory site awareness, search capabilities in Active Directory, and the capability to connect to any domain controller to change passwords rather than being required to connect to the PDC emulator.

Some features are missing from the Directory Services Client. It does *not* give Kerberos or Group Policy support, IPSec support, dynamic DNS support, or user principal name authentication capabilities. Unfortunately, if these features are needed, then an upgrade to Windows 2000 Professional is required.

The Directory Service Client Software for Windows 95/98 is included on the Windows 2000 CD. It can be found in the \clients\win9x folder. The Windows NT 4.0 Directory Services Client can be downloaded from the Microsoft Web site, and requires Service Pack 6a and Internet Explorer 4.01 or higher.



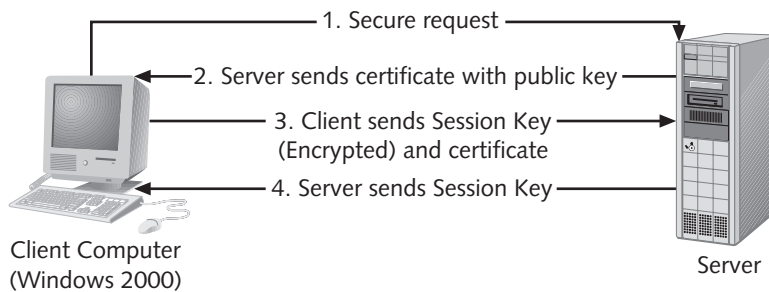
NTLM version 2 authentication can be implemented on a Windows NT machine by installing Service Pack 4.0 or later.

## Certificate-based Authentication

One of the limitations of an authentication protocol such as Kerberos is that it is based on a shared secret. In most cases, the shared secret is a password that the user knows, which is also stored in the domain database. If the user enters the correct password, the domain controller assumes that the user is who she says she is, and the user is given access to network resources. In this type of authentication system, every user that is going to log on to the network must have a user account with the associated password. This type of authentication works well if all of the users work for your company and have the appropriate user accounts. However, in many cases, you may want to authenticate users who do not have user accounts in your domain. A common example is an e-commerce site where the users of the site may be located anywhere in the world. In a distributed environment like this, a certificate-based authentication system is needed.

Rather than using a shared secret, **certificate-based authentication** uses certificates—digital information that contains the identification of the certificate holder, as well as public and private keys of the user. The advantage of using certificates is that a client that does not have a user account can authenticate and gain access to resources on the network. Rather than checking for a shared secret, the servers check the validity and trustworthiness of the certificates.

Authentication methods that take advantage of certificates include such protocols as Web browser Secure Sockets Layer (SSL), Encrypted File System (EFS), and IP Security (IPSec). The most common example is SSL, which is used when a client connects to an Internet Web server, and all the communication between the Web server and the client must be encrypted. A typical authentication exchange is illustrated in Figure 3-2.



**Figure 3-2** Certificate-based authentication

Authentication exchange involves the following steps:

1. A request is made by the client to initiate a secure authentication and transmission of data. The client makes this request by entering “Https” rather than “Http” as the protocol.
2. The server sends to the client a response consisting of the server’s assigned certificate and public key.
3. The client generates a random session key and encrypts it using the server’s certificate and public key. The session key is then sent back to the server, along with the certificate and public key previously assigned to the client.
4. The server will then use the client’s certificate and public key to send back the negotiated session key. The client can complete the authentication, and then the server and client use the session keys to encrypt all traffic between the two computers.

Setting up certificate-based authentication requires a complete infrastructure of servers to manage the certificates. This infrastructure is called a **Public Key Infrastructure (PKI)** and consists of a hierarchy of servers designated as **Certification Authorities (CAs)**. These CAs are used to issue and maintain the certificates that are handed out to clients.



Setting up a PKI is an important part of your overall security plan. Additional information on planning and implementing a PKI, and how the PKI can be used in your security plan will be discussed in Chapter 5, “Implementing a Public Key Infrastructure.”


## Remote Access Authentication

3

In addition to LAN authentication, users must also be able to authenticate remotely. Windows 2000 Routing and Remote Access incorporates various authentication protocols for users connecting either through a dial-up Remote Access Server (RAS) connection or through a Virtual Private Network (VPN) connection. Table 3.1 describes the available authentication options.

Table 3-1 Remote Access authentication

Protocol	Information
Password Authentication Protocol (PAP)	Passwords are sent in clear text, which makes this the least secure option. PAP is used mainly by non-Windows clients.
Challenge Handshake Authentication Protocol (CHAP)	The client's password is used to create a one-way hash challenge string to send to the server. The server also hashes the stored password and compares the two results. The password is never sent across the network in clear text. CHAP can also be used for non-Windows clients.
Microsoft Challenge Handshake Authentication Protocol 2 (MS-CHAP version 1 and version 2)	This is Microsoft's implementation of CHAP, which adds the ability to perform two-way authentications and uses separate keys for sending and receiving.
Extensible Authentication Protocol (EAP)	EAP allows future development of authentication methods such as smart card and biometrics.
Digest Authentication	This protocol can be used as an alternative to Integrated Authentication on an Internet Information Server. It uses a challenge/response concept, but may not be compatible with all Web browsers.

The remote access authentication protocols and the security issues related to remote access will be covered in detail in Chapter 8, “Securing Access for Remote Access Users.”

## ACCESSING RESOURCES

The process of authentication is used to give people access to a Windows 2000 network. If a user knows the right shared secret, or has the right certificate, the user is given access to the network by the authenticating server. However, the fact that a user has access to

the network does not mean that the user should have access to all resources on the network. The next step in understanding Windows 2000 security is to understand resource authorization. **Resource authorization** is the method used to secure network resources by assigning various levels of permission settings to users on the network.

## Security Principals

To understand how resource authorization works in Windows 2000, you have to begin with security principals and security identifiers. On a Windows 2000 network, any object that can be granted permission to access resources on the network is called a **security principal**. Security principals can be users, groups, computers, and network services.

Every security principal is assigned a **Security Identifier (SID)**. This SID is assigned to the security principal when the object is created, and the SID can never be changed. The SID is guaranteed to be unique. The SID is made up of two parts: the first part is a domain identifier, and all security principals in a domain will have the same domain identifier; the second part is the **Relative Identifier (RID)**, which is unique for each security principal. The SID is extremely important when setting any kind of security for resources on a Windows 2000 network. Although you grant permissions based on the user's display name, the operating system uses the SID to control access. When a user tries to access a resource on the network, the computer uses the user's SID to check permissions, rather than the person's name. This means that, if a user object name is changed, the permissions granted to the user do not change. However, if a user object is deleted and then recreated with the same name, the user will not be able to access the same resources because the SID is different.

## Access Tokens

In addition to SIDs, the concept of access tokens is also important. During a successful logon, the system creates an **access token** for the user. The access token contains:

- The user's primary SID
- The SIDs of any groups to which the user belongs
- The user's privileges and rights

The access token is used by the system whenever a user tries to access a resource. The token is presented by the operating system to any thread or application that requests security information before allowing access. For example, when a user tries to access a mailbox on an Exchange server, the access token is presented to the Exchange server so that the user can open their mailbox. When the access token is presented, the requested application will compare the SIDs on the access token to the permissions applied to the actual resource or service.

## Access Control Lists

These permissions on the objects are stored in **Access Control Lists (ACL)**. Every object in Active Directory or on an NTFS partition has an ACL, which is a listing of who has access to the object, and what level of access the security principal has. When a user tries to access one of these objects, the user's access token is presented to the security subsystem, and the subsystem compares the information on the access token to the ACL that is associated with the object. For example, when a user tries to access a file that is located on an NTFS partition on a file server, the user's access token is sent to the file server. The file server examines the SIDs that are part of the access token and then compares the SIDs with the ACL on the file. The user is then granted permission to the file based on the level of access the SIDs are given in the ACL.

Two types of access control lists can be configured on an object: **Discretionary Access Control List (DACL)** and **System Access Control List (SACL)**.

### Discretionary Access Control List (DACL)

The DACL lists the security principals that have been assigned permission to the object, as well as the level of permissions. Each entry in the list is called an **Access Control Entry (ACE)**.

### System Access Control List (SACL)

The SACL lists the security principals whose access to the resource needs to be audited. The list of ACEs indicates who is to be audited and the level of auditing required, such as successes or failures.

All of these components work together to determine a user's access to a resource. As an example, suppose a user, named Jim, is a member of the Managers group and the Sales group. When Jim logs on, his access token includes his personal SID, as well as the SID for each of the groups he belongs to. When Jim tries to access a file called Salesdata.doc on a file server, Jim's computer sends the access token to the file server. The security subsystem on the file server then examines the ACL on the file. Each ACE in the ACL grants permissions to a particular user or group, so each ACE is examined. In this case, suppose that one ACE grants read-only permission to Jim's SID, another ACE grants change permission to the Sales group, and a third ACE grants full control to the Managers group. Based on the entries in the ACEs, Jim will be granted full control permission to the file. If the SACL on the file indicates that all successful changes made to the file by anyone should be logged, then the security subsystem will log every change that Jim makes to the file.



If you are wondering how Kerberos authentication and resource authorization work together, the answer is found in the session ticket. The session ticket that a user gets from the KDC before accessing a resource includes the access token for the user.

## Editing Access Control Lists

Access Control Lists can be edited for both Active Directory objects and NTFS objects. To control access to Active Directory objects, use the Active Directory Users and Computers tool.

To edit an Active Directory Access Control List:

1. Click **Active Directory Users and Computers** from the Administrative Tools menu.
2. Click the **View** menu, and then click **Advanced Features**. This will make the Security Tab available for all objects in Active Directory.
3. Right-click the Active Directory object that is to have its permissions edited, and then click **Properties**.
4. Click the **Security** tab.
5. The access control list will appear listing all Access Control Entries (ACEs) that have already been assigned permissions to the object. Figure 3-3 shows what the ACL looks like on an Organizational Unit.
6. To add or remove an ACE, click the **Add** or **Remove** button at the top rightmost corner of the dialog box.

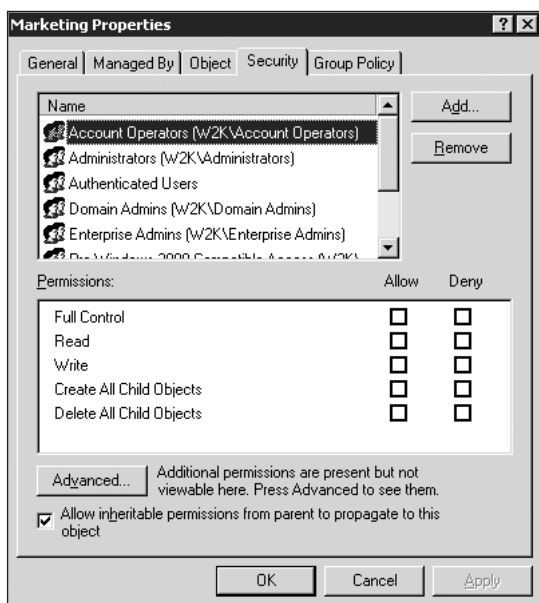
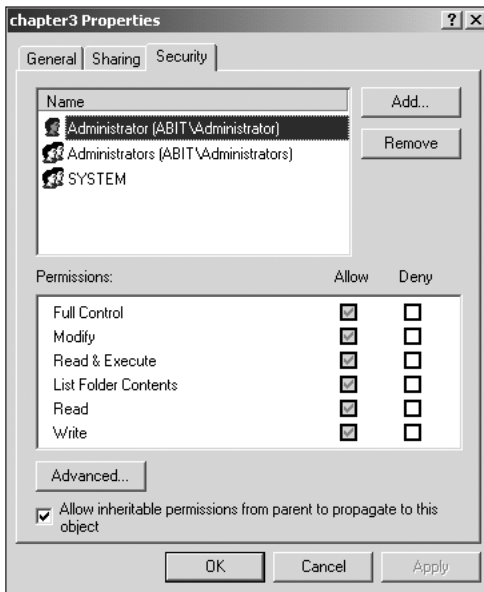


Figure 3-3 Active Directory Access Control List

The permissions on NTFS objects can be edited in the same way as Active Directory objects. The following steps illustrate how to edit an NTFS Access Control List.

To edit an NTFS Access Control List:

1. Right-click the NTFS folder or file that is to have its permissions edited, and then click **Properties**.
2. Click the **Security** tab.
3. The access control list will appear listing all Access Control Entries that have already been assigned permissions to the object. Figure 3-4 shows an example.
4. To add or remove an ACE, click the **Add** or **Remove** button at the top rightmost corner of the dialog box.



**Figure 3-4** NTFS Access Control List

Both the Active Directory and NTFS ACL dialog boxes have advanced editing features that can be applied to the list. Some of the advanced features available include more advanced permissions for the objects, auditing selections, and ownership information. (See Figure 3-5 for an example.) Advanced features will be discussed in the next section.

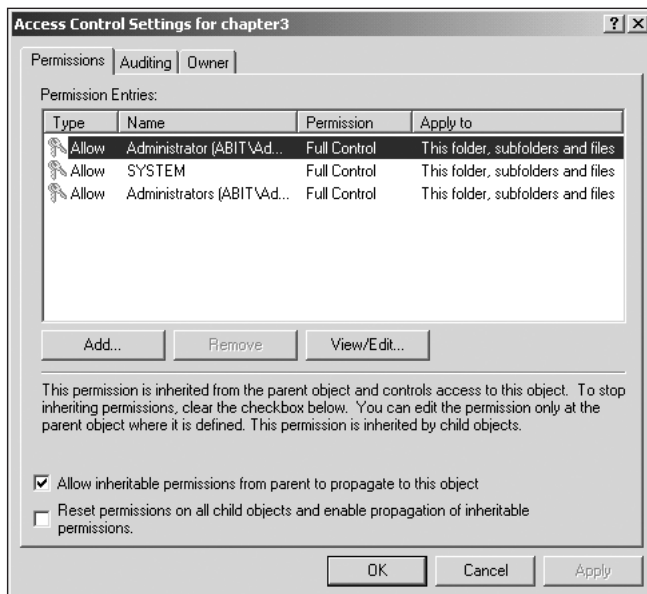


Figure 3-5    Advanced ACL settings

## SECURING FILE RESOURCES

For many companies, the data that is stored on their computers is the most valuable commodity that the company owns. This data must be protected, while at the same time, it has to be available to the people who need it. One of the most challenging aspects of an administrator's work is to ensure that all the people who need access to the data have the correct level of access. Windows 2000 provides two methods of controlling access to this data: **share permissions** and NTFS permissions.

### Share Permissions

Before files can be accessed across a network connection, the folder containing the files must be shared. To share a folder means that people are given access to the information inside that folder across a network. Administrators must spend time to ensure that the shared file structure is designed so that access to the files is efficient and easy to manage.

Here are some ways to create shares in Windows 2000:

- Through Windows Explorer
- Through the Create Shared Folder Wizard
- Through the Share Folder tool in the Computer Management snap-in

Each of these tools can be used in different ways. Windows Explorer can be used to create shares and fully configure the shared directory. The Create Shared Folder Wizard provides a simplified, Wizard-driven method for creating shares. The Shared Folder tool is powerful because this tool can also be used to monitor who is using the file, as well as to create and set share permissions on folders.

To create a shared folder using Windows Explorer:

1. Right-click the folder to be shared and click **Sharing**.
2. Click **Share This Folder**. Type in the share name or accept the default. If your network still has DOS or Windows 3.x client computers, remember to limit the share name to eight characters or less.
3. Click the **Permissions** button to display the share permissions associated with a folder.
4. In most cases, the default permission of Everyone having Full Control is inappropriate. To remove the Everyone group from the list of users, make sure the group is selected and click **Remove**.
5. To add other users or groups, click **Add**, and select the users or groups that you want to give permissions to the folder.



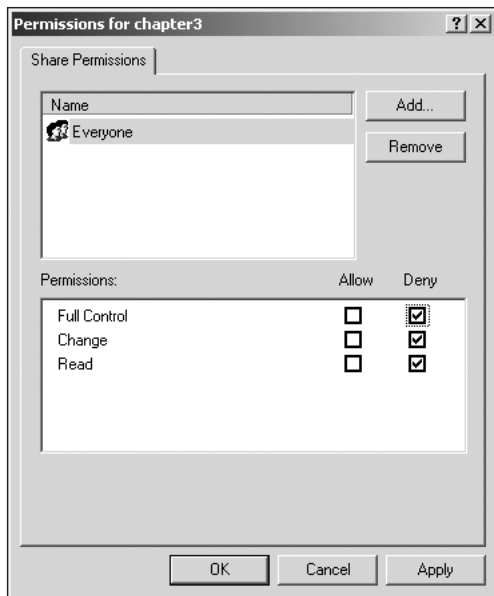
It is recommended that the Everyone group be removed from all shares and NTFS permissions, as it is not appropriate and not required. Most domains should be using specific accounts or groups, not generic accounts (guest) or groups (everyone).

Windows 2000 supports only three share permissions as shown in Table 3-2.

**Table 3-2** Share permissions in Windows 2000

Windows 2000 permission	Permissions granted
Read	Allows the user to browse the file and folder names, including subfolders, read the data in a file, and execute programs.
Change	Includes the Read permission, plus permission to add and delete files and subfolders, and change the data in the files.
Full Control	Includes the Read and Change permission, plus the right (on an NTFS partition) to change permissions on a folder or file and to take ownership of a folder or file.

Notice that Windows 2000 does not have a *No Access* share permission. Instead, to deny a user or group access to a shared folder, an administrator must explicitly deny the user permission.



**Figure 3-6** The Everyone group explicitly denied permission to a share

## Windows 2000 Shared Folder Default Properties

When a new share is created, the default permission on the share gives the Everyone group full access. One of the first steps an administrator should perform after creating a share is to remove this permission and substitute more appropriate permissions.



The Everyone group includes all users who have access to the computer, whether or not members of the domain.

When a share is created and a user is assigned permission to that share, the user will also have the same level of permissions to all subfolders inside that share. In other words, permissions are inherited by subfolders.

Shared permissions are cumulative for the user. All the permissions assigned to the user and any group of which the user is a member are combined, and the least restrictive of all the permissions applies. For example, a user named Jill is a member of the Sales group, as well as a member of the Managers group. If the Sales group is assigned Read permission to a share, the Managers group is assigned Full Control, and Jill is assigned Change permission, Jill will have Full Control of the share.



## The Shared Folder Snap-in

The **Shared Folder snap-in** is a Windows 2000 tool that can be used to perform the following tasks:

- Create shares, monitor access to shares, and remove shares from local and remote servers
- Change share permissions and limits
- Send messages to specific computers, or throughout the network
- Monitor user sessions
- Monitor and close shared files

The Shared Folder administration tool is part of the Microsoft Management console (MMC), or it can be added as a standalone snap-in to any custom MMC.

### Creating A Shared Folder Using The Shared Folder Snap-In

The Shared Folder snap-in provides a Create Shared Folder Wizard that guides the administrator through the process of creating shared folders. To start the Wizard, right-click Shares in the Shared folder snap-in and select Create a new share. The Wizard prompts for the physical location of the shared folder and the share name, and provides a choice of what permissions to assign. The default share permission is the Everyone group with Full Control. These share permissions can be replaced by one of three standard share permissions:

- Administrator has Full Control; users have Read access
- Administrator has Full Control; all other accounts have no access
- Customized permissions

Share and folder properties can be changed at any time by accessing the properties sheet of the shared folder.

### Monitoring Who is Accessing a Share

One of the valuable options available through the Shared Folder snap-in is the option to monitor who is accessing a share. This can be useful if you need to shut down a server and would like to warn any users connected to the server that you are about to shut it down. To monitor a share, double-click Sessions in the Shared Folder MMC. The details screen will list all of the users who currently have a session open on the server, as well as the connection time and how long the connection has been idle.

Double-clicking the **Open files** option provides details about which files are being accessed across the network.

### Sending a Message to Users

Before shutting down the server, you might want to send users a message to notify them that you are about to do this. You can send a network message to all the users connected

to a share by right-clicking the Shared Folders snap-in, and selecting All Tasks, Send Console Message. Type the message in the space provided, and select the computers to which you want to send the message. The message can be sent to all connected computers, or the computers can be removed from the list. The message can also be sent to logged-in users, but each name has to be typed in.

## NTFS Permissions

The second way that access to file resources can be controlled is through **NTFS permissions**. When permissions are set on a shared folder, the permissions determine what restrictions apply to the folder when it is accessed from another computer across a network connection. However, NTFS permissions are applied whenever a file or folder is accessed, whether the person is logged on to the computer where the file is located, or the person is accessing the file across a network connection.

### NTFS Concepts and Rules

It is important to understand NTFS permissions and how they are applied:

- NTFS permissions are set on the Security tab, which may be accessed by right-clicking on any file or folder and selecting Properties.
- NTFS permissions are cumulative. If a user is a member of multiple groups that have different permissions, the final permission is the sum of all permissions. For example, a user named Jim may be a member of a group called Marketing, as well as a group called Sales. If Marketing is given Read permission to the folder, and Sales is given Full Control, then Jim will have Full Control access to the folder.
- The Deny Access file permission overrides all other permissions. In the above example, if Jim is explicitly denied Full Control rights to the folder through an individual assignment or through a different group assignment, this will override any permission the users may have. Deny Access overrides all other permissions because of the way the ACEs are evaluated by the security subsystem. When a user tries to access a folder, all of the ACEs that deny access are evaluated first. If the SID on any of the deny access ACEs matches the SIDs in the user's access token, the deny access permission is applied and no more ACEs are evaluated.
- NTFS folder permissions are inherited by child folders and files unless otherwise specified. Clearing the Allow inheritable permissions from parent to propagate to this object option on the Security property sheet can prevent inheritance of NTFS permissions.
- NTFS permissions can be set at a file level, as well as at a folder level.
- Unless explicitly specified, Full Control for Everyone is the default NTFS permission for all files.
- Windows 2000 has a set of standard NTFS permissions, as well as special permissions. Table 3-3 lists the standard NTFS permissions.

**Table 3-3** Standard NTFS permissions

Windows 2000 Permissions	Permissions Granted
Full Control	The user can make any changes to the file or folder. The detailed permissions are listed in Table 3-4.
Modify	Gives full permissions, except the permission to Delete Subfolders and Files, Change Permissions, and Take Ownership.
Read and Execute	Gives permissions to Traverse Folders, List Folders, Read Attributes and Extended Attributes, Read Permissions and Synchronize. These permissions are inherited by both files and folders.
List Folder Contents	Same as Read and Execute permissions, except that the permissions are inherited only by folders and not by files. Visible only on folders.
Read	Same as Read and Execute, except without the permission to Traverse Folder. Inherited by files and folders.
Write	Gives permissions to Create Files and Folders, Write Attributes and Extended Attributes, Read Permissions and Synchronize.

## Special NTFS Permissions

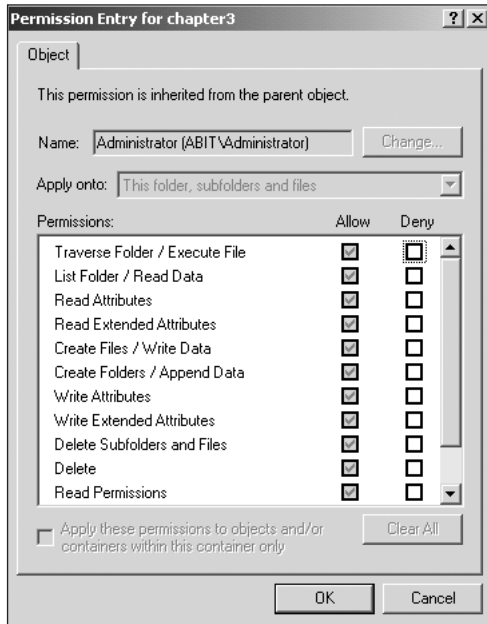
Windows 2000 uses 13 individual NTFS permissions to specify the level of access provided to a given resource. Occasionally one of the standard permissions may not provide detailed enough control. To access the special permissions, click the Advanced button in the Security dialog box on the Properties tab for the folder or file. The resulting Access Control Settings dialog box presents special permissions for existing accounts to be assigned.

To view or edit special NTFS permissions:

1. Click **Advanced** on the disk object's Security properties sheet.
2. Select the User or Group account.
3. Click **View/Edit**. The Permissions Entry dialog box opens.

The Permissions Entry dialog box provides for the selection of special permissions, as well as the rules for their application. Special permissions can be applied at the following levels:

- This folder only
- This folder, subfolders, and files (default)
- This folder and subfolders
- This folder and files
- Subfolders and files only
- Subfolders only
- Files only



**Figure 3-7** Permissions Entry dialog box

The following table shows the special access permissions that can be applied and their function.

**Table 3-4** Special access permissions

Special Permission	Function
List Folder/Read Data	Supports or cancels viewing of file names and subfolder names within the folder.
Traverse Folder/Execute File	Supports or cancels passing through folders that the user does not have explicit permission to enter, in order to get to an intended folder. For example, a user may not have permission to read the Salesdata folder, but may have Read permission to JuneSales.doc in the Salesdata folder. If the user has Traverse Folder permissions, the user would be able to open the JuneSales.doc file by typing in the full path to the file.
Read Attributes	Supports or cancels ability to read attributes of a file or folder.
Read Extended Attributes	Supports or cancels viewing of extended attributes of a file or folder. Extended attributes are additional information attached to a file, as defined by an application.
Create Files/Write Data	Supports or cancels the creation of files within the folder (applies to folders only). Supports or cancels the making of changes to the file and overwriting existing content (applies to files only).

**Table 3-4** Special access permissions (continued)

Special Permission	Function
Create Folders/Append Data	Supports or cancels the creation of folders within a folder (applies to folders only). Supports or cancels the making of changes to the end of the file, but not changing, deleting, or overwriting existing data (applies to files only).
Write Attributes	Supports or cancels changing the attributes of a file or folder, such as read-only or hidden. Attributes are defined by NTFS.
Write Extended Attributes	Supports or cancels changing the extended attributes of a file or folder. Extended attributes are defined by programs and can vary.
Delete Subfolders and Files	Supports or cancels the deletion of subfolders and files, even if the Delete permission has not been granted on the subfolder or file.
Delete	Supports or cancels the deletion of the file or folder.
Read Permissions	Supports or cancels the reading of permissions for the file or folder.
Change Permissions	Supports or cancels the changing of permissions for the file or folder.
Take Ownership	Supports or cancels the taking of ownership of the file or folder.

## Combining Share and NTFS Permissions

NTFS permissions are often combined with share permissions to provide a strong combination of local and remote security for files and directories. When share and NTFS permissions are combined, the following rules apply:

- When a user is accessing a share across a network, and the NTFS and share permissions are combined, the most restrictive permission is the overriding permission. For example, if the share level access is Full Control, but the NTFS permissions are set at Read, then the user will have Read permission.
- When a user accesses a file locally, only NTFS permissions apply.

## Shared and NTFS Permissions Example

To understand how NTFS and share permissions work together, consider the following example:

**Table 3-5** Permissions example

Users:		Group Membership:
User1		Sales, Marketing, Domain Users
User2		Managers, Domain Users
Directory Structure	Share Permissions	NTFS Permissions
SalesData	Sales—Full Control	Sales—Modify
	Managers—Read	Marketing—Full Control
CompanyData	Domain Users—Change	Default
HRData		Default

1. What permissions would User1 have to the SalesData directory if the user was accessing the folder across a network connection?
2. What permissions would User1 have if the user was logged on to the computer where the folder was located?
3. What permissions would User2 have to the Company Data folder if accessing the folder across the network?
4. What permissions would User2 have to the HRData folder if accessing the folder across the network? How could this folder be better protected if users should be allowed to view but not change the data in this folder?



One of the essential components to your security policy will be managing user access to network resources. Refer to the “Planning Best Practices” section at the end of this chapter for some suggestions on how to optimize the security of the resources, while minimizing the amount of administrative effort.

---

## ENCRYPTING FILE SYSTEM

Using NTFS permissions is one way of protecting the data that is stored on the local hard drive. However, if the hard drive is removed from the computer and put into another computer, anyone with administrative rights could access the data. To maximize the level of data protection, Windows 2000 also provides the option to encrypt files on the local hard disk, or **Encrypted File System (EFS)**.

If a security plan includes implementation of the encrypting file system, several decision points have to be considered. When a user encrypts a file, a private key is created and assigned to that user. The user then uses the private key to encrypt and decrypt the file. The private key is stored as part of the user’s profile, and only the user has access to the private key. Since encryption is based on this private key, only one user can read the encrypted file. In other words, public or shared folders must not be encrypted, because only the person who invoked the encryption will be able to access the information.

If the person who encrypted the files leaves the company, or if the user’s private key became lost, the data would be unrecoverable, because no one else would have the private key needed to decrypt the data. To address this problem, Windows 2000 requires the use of a **recovery agent(s)**. Someone must decide who will take on this role and be able to recover an EFS encrypted file, in the event that the original private key is not available.

On a standalone Windows 2000 computer that is not a member of any domain, the initial administrator account is, by default, assigned the role as the EFS recovery agent. If the computer is a member of a domain, the administrator account of the first domain controller installed in the domain is assigned the role as the EFS recovery agent.

## The File Encryption Process

The following steps outline how a file is encrypted using EFS:

1. When a file is encrypted, a file encryption key is automatically generated and is used to encrypt the data. Two additional header fields are added to the document. The **Data Encryption Field (DDF)** contains the file encryption key that will be used to decrypt the document. The **Data Recovery Field (DRF)** also contains the file encryption key that the recovery agent can use to decrypt the data if necessary.
2. The file encryption keys, which are stored in the DDF and DRF headers, are encrypted using the user's and recovery agent's public encryption keys, respectively. This ensures that only the user that has the matching private key can decrypt the file encryption key used to decrypt the file itself.
3. To decrypt the file, the file encryption key stored in the DDF is decrypted using the user's private key. Only the user whose public key was used to encrypt the keys has the right private key. If the recovery agent is attempting to decrypt the data, the agent's private key will decrypt the file encryption key stored in the DRF header.
4. The file encryption key is then used to decrypt the actual data.

## Implementing EFS

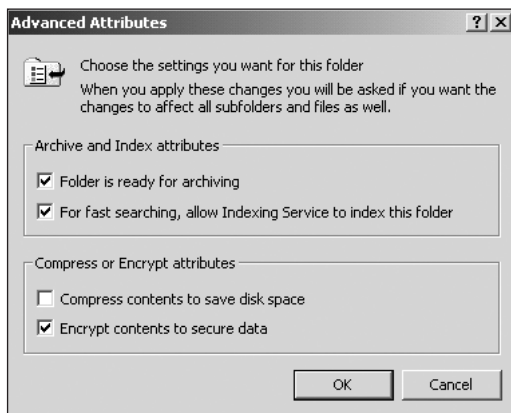
For the most part, once file encryption has been enabled on a folder, the actual process of encryption and decryption will be transparent to the user. To encrypt a file or folder, use the following procedure:

1. Right-click the file or folder to be encrypted. Click **Properties**.
2. Click the General tab, and then click the **Advanced** button.
3. In the Advanced Attributes dialog box, click the check box that states **Encrypt contents to secure data**. Figure 3-8 illustrates this option.

When the user saves the file, it is automatically encrypted; when the user reads the file, it is automatically decrypted if the user has the correct private key.

There are some limitations to the transparency, however. Encrypted data cannot be shared across a network (because only one person has the key to decrypt the data), and only data stored on NTFS volumes can be encrypted. Users can encrypt data to be stored on a server; however, it will not be encrypted as it is sent across the network to the server. To encrypt the network traffic, use an option like IPSec.

Only files are encrypted, not folders. However, folders can be configured so that all files in the folder will be encrypted. When encrypted files are moved, copied, or backed-up to a backup device, they remain encrypted. This applies even if a file is moved to a folder that is not set for encryption.



**Figure 3-8** Advanced Attributes dialog box

The management of the keys and certificates needed for EFS is also, by default, transparent to the administrator. When a user encrypts a file for the first time, a File Encryption Certificate, including the public key and private key, is issued to the user automatically. This encryption certificate and the associated keys are stored on the local computer hard disk and are not accessible to any other user.

## Managing Data Recovery

It is important to be able to recover encrypted data (for example, when the hard drive, where the private key is stored, crashes). Windows 2000 has provided a recovery policy that will always make it possible for the data to be recovered.

To ensure that no file will ever be lost permanently, Windows 2000 does not allow users to encrypt any files unless a recovery policy is in place. By default, the administrator of a stand-alone machine and the administrator account on the first DC created in a domain will be the recovery agents for the network. This means that these accounts also have a certificate issued to them that includes a private key that can be used to decrypt any encrypted file on the computer. The default policy can be changed through group policies.

To recover data by decrypting another user's files:

1. Make sure that you are logged in as a user who has data recovery agent rights.
2. Right-click the file that you need to recover and click **Properties**.
3. Click **Advanced** and clear the **Encrypt contents to secure data** check box. Click **OK**, and then click **OK** again.
4. Open the file.

If the file that is being recovered is confidential and needs to be encrypted again after it has been decrypted by the recovery agent, the administrator may give another user the right to take ownership of the folder. That user can then encrypt the file with his or her key.





The EFSinfo utility, which is included in the Microsoft Windows 2000 Server Resource Kit, can be used to see which private key is required to decrypt an EFS encrypted file.



In order for the recovery agent to be able to decrypt encrypted files, the data recovery certificate must be moved to the hard disk of the computer with the encrypted files. By default, the certificate used by a domain data recovery agent is located only on the domain controller where the certificate was first created. You can use the Certificates MMC snap-in to export the certificate, and then import the certificate to the computer where it is needed. After you have recovered the files, make sure that you remove the certificate from the computer to ensure that no one else can get access to the data recovery certificate.

## SECURING PRINTERS

In addition to securing access to data stored on the network, it is equally important to secure the access to printing the data. Securing the transmission of the data being sent to the printer and the physical location of the printer are often overlooked. Windows 2000 provides several enhanced features to assist in printer security.

Printer security, similar to file security, also consists of assigning Access Control Entries (ACEs) to Access Control Lists (ACLs). The permissions available on printer objects are listed in Table 3-6.

**Table 3-6** Printer permissions available in Windows 2000

Windows 2000 Permissions	Permissions Granted
Print	Permission to submit print jobs
Manage Printers	Includes Print permissions and also allows the user the ability to share a printer, and change the printer's properties
Manage Documents	Permission to change the order, pause, or delete documents in the print queue



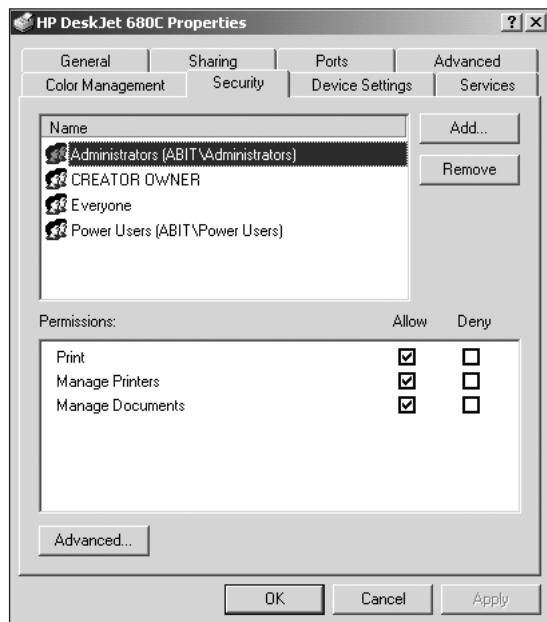
A special group called Creator Owner is assigned the Manage Documents permission, by default. This allows users to manage their own print jobs when printing to a print device.

## Setting Printer Security

In order to configure security on a printer, use the following procedure:

1. Click **Start**, **Settings**, and then click **Printers**.
2. Right-click the **printer icon** and click **Properties**.

3. Click the **Security** tab to view the ACL. Figure 3-9 shows an example.
4. Click the **Add** or **Remove** button to edit the Access Control Entries.
5. Check the level of permissions needed.



**Figure 3-9** A printer Access Control List in Windows 2000

The physical location of the print device should also be addressed when planning printer security. Documents are often printed and left at the printer location. Other users may also be printing to the same printer and could possibly view confidential information. Print devices that are used to print confidential data should be located in a secure place. The administrator can also edit the Access Control List so that only authorized people can print to a specific printer.

Another possible security problem may lie in the transmission of the print job to the printer. Any user capable of using a network sniffer can easily capture the print data and view what is being sent to the printer. Windows 2000 now includes a new security feature called Internet Protocol Security (IPSec). This new security protocol gives administrators the ability to encrypt the packets from the source workstation to the print server. IPSec will be discussed in Chapter 7, “Securing Network Communications.”

---

## SECURING THE REGISTRY

The Registry is an integral part of the Windows 2000 operating system. Since most Windows settings reside in the Registry, it only makes sense to provide a way of securing

and controlling access. Applying security to the Registry is almost identical to controlling NTFS file system access. Permissions to the Registry information are based upon an Access Control List and a combination of possible permissions. If a new key is created under an existing key, the newly created key will inherit any applied permissions from the existing key. Permission inheritance is also similar to NTFS permissions, in that if any permission is changed at a higher lever, it can also be reset and inherited to the lower-level keys.

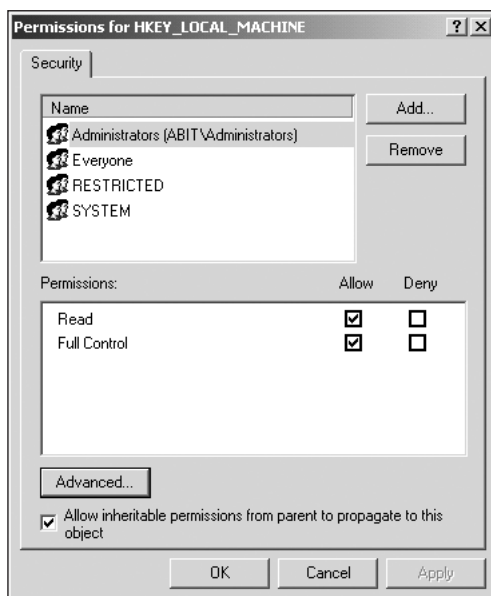
Table 3-7 lists the standard Registry permissions. Figure 3-10 shows the standard access control list on a registry key.

**Table 3-7** The standard Registry permissions in Windows 2000

Registry Permission	Permissions Granted
Read	Allows user to read, identify, query, and be audited within selected subkeys
Full Control	Gives the user full permissions on selected subkeys



Be careful of entries that are listed only as Read access. There may be special permissions applied to the user. There is no indication of special permissions being applied unless you click the Advanced button and view the special permission entries.



**Figure 3-10** The Registry Access Control List

When you click the Advanced button, a combination of the following special permissions can be applied to control registry access. Table 3-8 lists the special permissions. Figure 3-11 shows the special permissions assigned to a Registry key.

Table 3-8     The Special Registry permissions

Registry Permissions	Permissions Granted
Query Value	Search for the settings of a value entry in a subkey
Set Value	Change a value in a subkey
Create Subkey	Create a subkey
Enumerate Subkeys	List all subkeys within a key or subkey
Notify	Set auditing on keys or subkeys
Create Link	Link this key to other subkeys
Delete	Delete selected key or subkey
Write DAC	Modify the DACL for the key
Write Owner	View or take ownership of the selected key
Read Control	Read security information within the key

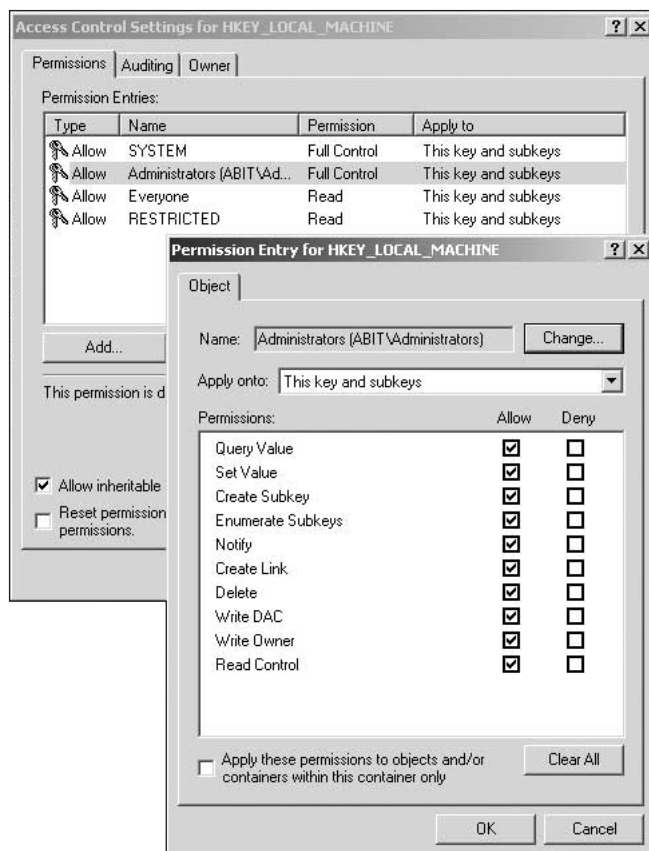


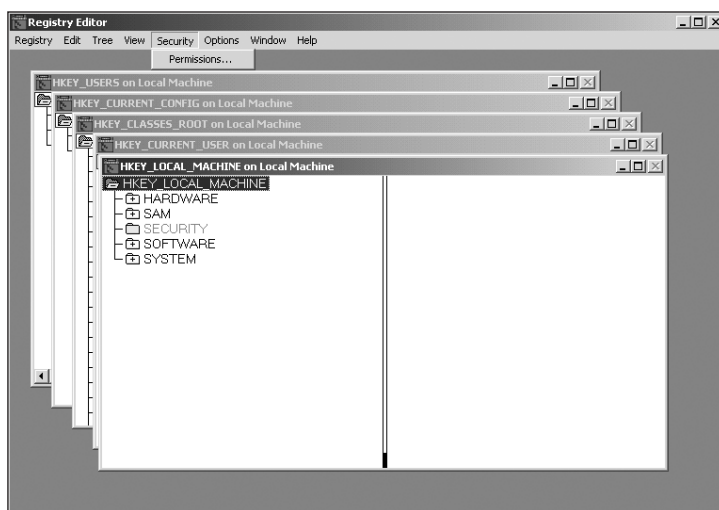
Figure 3-11     Special permissions applied to the HKEY\_LOCAL\_MACHINE key

You can modify the Registry Discretionary Access Control Lists (DACLS) by using the Registry Editor utility. To modify the permissions to a Registry key using the Registry Editor, use the following procedure:

1. Click the **Start** button.
2. Click **Run**.
3. Type **Regedt32**.
4. Select the Hive and Subkey on which security is to be modified and then click the **Security** menu and click **Permissions**. Figure 3-12 shows an example.



Always test any Registry changes on an off-line test machine before deploying in a production environment. A backup of the current Registry should also be available before making any changes. Incorrect changes to the Registry can potentially crash the operating system.



**Figure 3-12** Modifying permissions using the Registry Editor

## Default Registry Settings

Understanding the default registry settings will help the administrator plan Registry access, as well as troubleshoot any associated problems. Default Registry settings includes the following:

- Full Control is assigned to Administrators and the System Account.
- The Everyone group is assigned Read permissions to most hive roots, but not to subkeys.

- The Power Users group can create subkeys in the HKEY\_LOCAL\_MACHINE\SOFTWARE key. This gives the Power User the ability to install software on the computer.

---

## CONFIGURING AN AUDIT POLICY

In addition to securing network files and resources, another major task for the administrator is to analyze and verify that the security policy is effective. **Auditing** lets the administrator track processes related to security, the network, and users by utilizing the Windows 2000 Event Viewer utility. If a security breach does occur on the network, auditing can help discover how the event occurred and the extent of the breach, as well as provide evidence of the occurrence. The administrator chooses what processes and actions are to be audited, and then uses the audit information to detect any security problems, usage pattern statistics, and network trends.

An audit policy should be designed to reflect the company's security and tracking goals, and to ensure that the amount of logging does not overburden the Event Viewer logs. It is easy to configure auditing to include many options, only to find that the event logs have filled up with hundreds of events that do not need to be tracked. This situation makes finding important events difficult and increases the overall size of the event logs. Auditing can also have an impact on system performance, and so it is crucial to choose only events that are important to the goals of the security policy.

Auditing security events is a two-step process. First, auditing must be enabled using the security policy MMC of the workstation or domain. Second, if object access auditing, such as file or printer access, is needed, then auditing must be set on each object's System Access Control List (SACL).

When Windows 2000 is first installed, no auditing policies are set. The administrator has to set up the auditing policy and choose what will be audited. Table 3-9 lists the categories that can be audited.

**Table 3-9**    Auditing categories in Windows 2000

Event Category	Explanation of Event
Audit Account Logon Events	Activated when a user logs onto a computer. If the logon occurs on the local computer, the event is recorded on the local computer's event log. If the logon is on a Domain Controller, the DC will record the event. This includes issuing Kerberos tickets for resource access.
Audit Account Management	Activated whenever a user or group is created, deleted, or modified. This category also tracks successful or unsuccessful password changes.
Audit Directory Service Access	Activated when an Active Directory object is accessed. The specific Active Directory object that is to be audited must also have auditing enabled.

**Table 3-9** Auditing categories in Windows 2000 (continued)

Event Category	Explanation of Event
Audit Logon Events	Activated when a user logs on or off a local computer or Active Directory. Audit logon failures to find out if password hacking is taking place.
Audit Object Access	Activated when an object such as a folder or printer is accessed. The administrator must also configure the specific object for audit successes and failures.
Audit Policy Change	Activated when a policy that affects security, user rights, or auditing is changed.
Audit Privilege Use	Activated whenever a user uses an assigned right, such as changing the system time or taking ownership of a file.
Audit Process Tracking	Activated any time that an application process takes place. Can assist developers in discovering which files or Registry settings an application accesses when executing a command.
Audit System Events	Activated when a system event, such as the computer rebooting, takes place.

As mentioned earlier, there are two steps to set up an audit policy. The first step is to configure the categories that are needed to meet the auditing goals. The following steps illustrate how to configure these categories:

1. Click **Active Directory Users and Computers** from the Administrative Tools menu. If auditing is to be configured on a standalone computer, click **Local Security Policy** from the Administrative Tools menu.
2. In Active Directory, right-click the Domain or OU that is to have the configured audit policy applied. To have the Domain Controllers audited, right-click the **Domain Controllers OU**. Click **Properties**.
3. Click the **Group Policy** tab and then the **Edit** button. If there is no group policy to edit, choose **New** to create a new policy.
4. In the left pane of the group policy screen, maneuver to Computer Configuration, Windows Settings, Security Settings, Local Policies, Audit Policy. Figure 3-13 shows the auditing categories.
5. Double-click the event that is to be audited.
6. In the Security Policy Setting dialog box, click **Define these policy settings**, and choose whether to audit successes, failures, or both. Figure 3-14 illustrates the options.

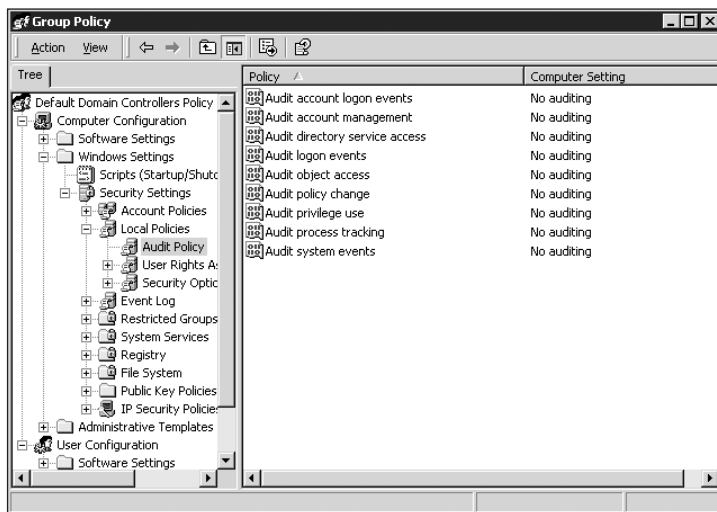


Figure 3-13 Audit categories available in Windows 2000

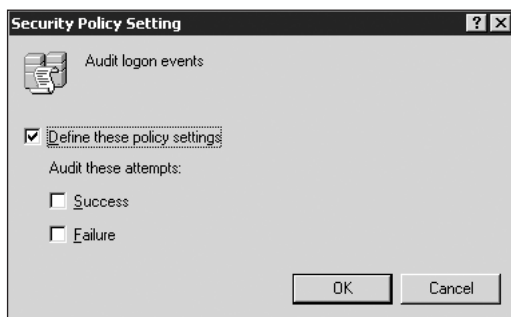
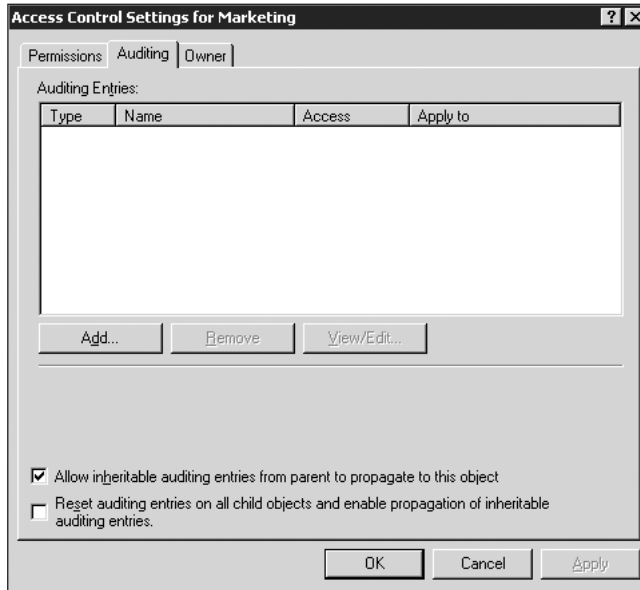


Figure 3-14 Configuring the security policy setting

The second step to enable auditing is required only if the auditing policy is to include the auditing of object or directory service access. If this is the case, the object itself must have auditing entries assigned to it. To enable this, use the following procedure:

1. Right-click the object that is to be audited and click **Properties**. The object can be a folder, file, or Active Directory object.
2. Click the **Security** tab.
3. Click the **Advanced** button and then click the **Auditing** tab. Figure 3-15 shows the Auditing tab.
4. Click **Add** to add a new group or user to be audited on this object. Adding a new group or user means that when these users access the object, the access will be logged. If you want to audit all access to an object, select the **Everyone** group, and add it to the Audit list.





**Figure 3-15** Enabling auditing on a folder object

5. In the Auditing Entry dialog box, choose which objects should be audited from the **Apply onto** drop-down menu.
6. Select the **specific types** of access auditing events and choose to audit **successful events** or **failed events**, or **both**. Figure 3-16 shows the auditing options.

## Managing Event Logs

When auditing has been enabled, all the audited events are recorded in the Windows 2000's Security Event Logs. If the level of logging is set at a high level, the number of events in the security logs will increase rapidly, so it is a best practice to view the logs regularly. As with many security options, deciding what to log is a balancing act between logging so much information that you never have time to analyze all the security events, and logging too little information so that you miss an important security event. In most cases, you should start by logging at a fairly high level, determine which information is not relevant, and then remove that information from the logging policy.

To view the Event Logs, open the Event Viewer from the Administrative Tools menu. The Application Log, Security Log, and System Log are available on all Windows 2000 computers. The Directory Service, DNS Server, and File Replication Service logs are available only on domain controllers that are also DNS servers. Figure 3-17 shows an example of the Event Viewer on a Domain Controller.

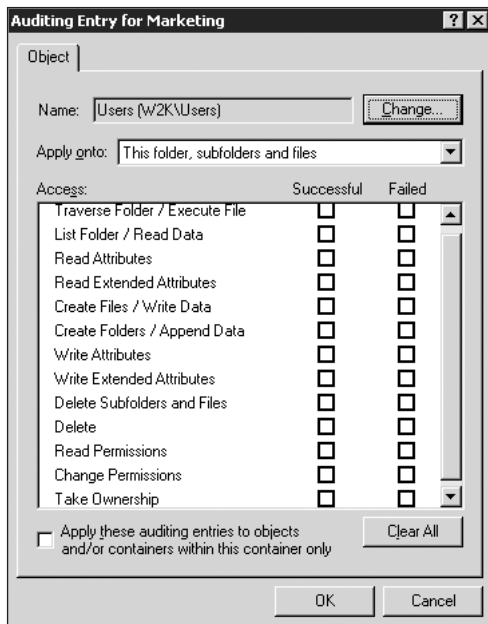


Figure 3-16 Configuring object auditing options

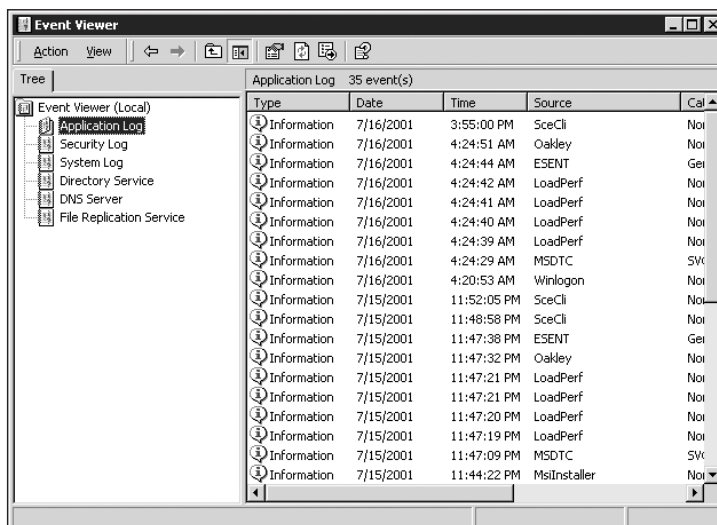
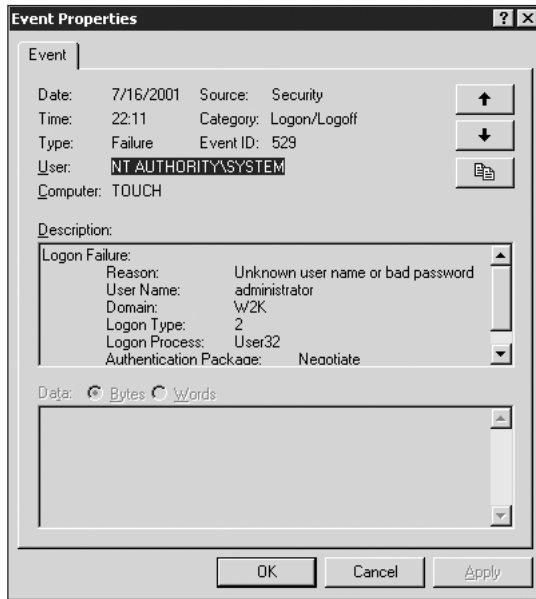


Figure 3-17 The Event Viewer on a Domain Controller

A specific event can be viewed by double-clicking the entry. The Event Properties dialog box will appear and list various types of information. This information includes data such as the date and time of the event, the event category, the user and computer name

that invoked the event entry, and a description of the event. The Event ID number can be researched on the Internet to help determine the meaning of certain event entries. Figure 3-18 shows an example of a single event.



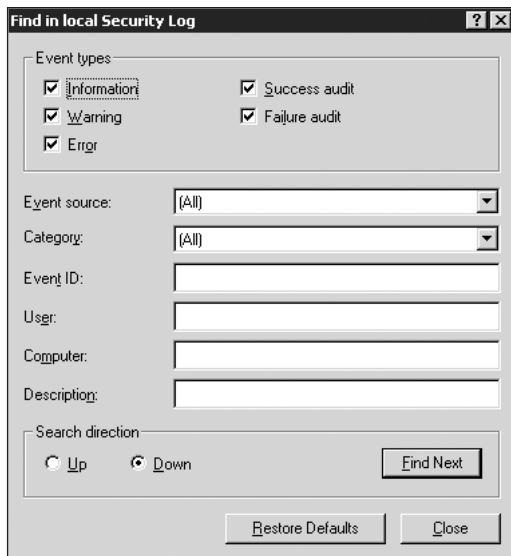
**Figure 3-18** The Event Properties dialog box

Event logs can easily become overburdened with information, making it difficult to find specific types of events. Each log in the Event Viewer includes a Find feature that allows searches based upon the individual attributes that are recorded in an event entry. Figure 3-19 illustrates what attributes can be searched for.

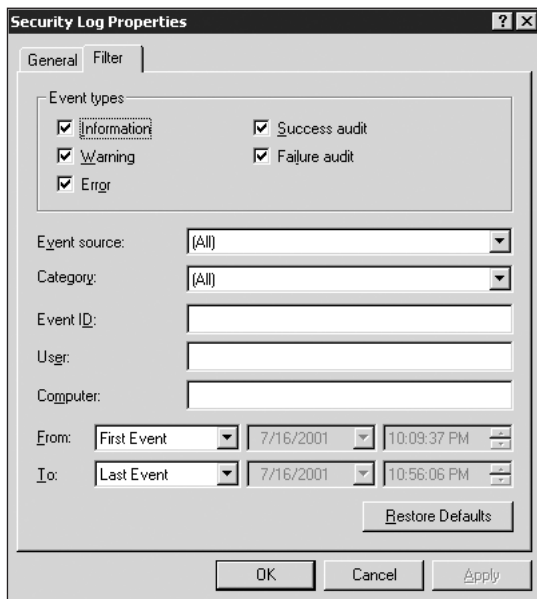
To search for a specific event, select the log that is to be searched, and choose **Find** from the View menu.

Another method that can be used to find a specific event is to **filter** an event log. Filtering allows the administrator to view only the events that are important and hide any events that are not important. To filter an event log, use this procedure:

1. Click **Event Viewer** from the Administrative Tools menu.
2. Right-click the log that you want to filter. Choose **Properties**.
3. Click the **Filter** tab. Figure 3-20 shows an example.
4. Choose the event types or any other filtering options (such as Event Source, Category, etc.) that are needed to filter the log. Then click **OK**.
5. To revert back to the unfiltered view, return to the filter tab and click **Restore Defaults**.

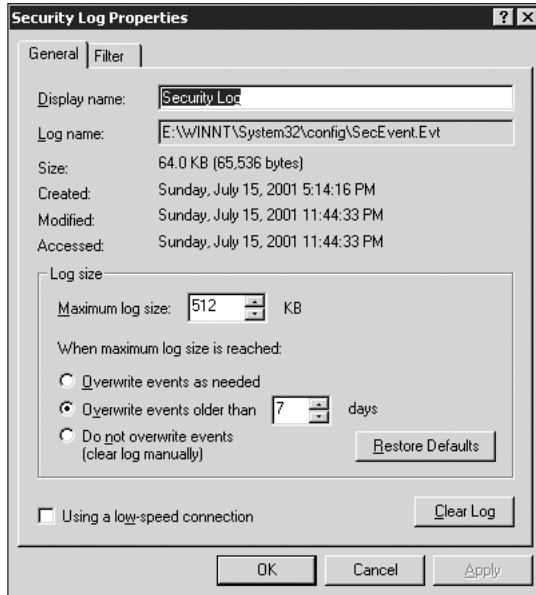


**Figure 3-19**    The Find dialog box



**Figure 3-20**    Filtering the event log

Each event log has general properties that can be configured, including Maximum log size and Overwrite events options. To view the Properties, choose a log in the Event Viewer, right-click, and click Properties. Figure 3-21 shows an example.



**Figure 3-21** Event Log properties

Event Logs can also be used to track system trends. To be able to compare entries for an extended amount of time, you may need to archive or save the event logs before they become overwritten. To archive an Event Log, select the log, right-click, and click **Save Log File As**. You are also given the choice to save the log file when you click **Clear All Events** from the context menu.

## PLANNING BEST PRACTICES

Planning is an essential part of securing resources on Windows 2000 networks. Some important best practices are listed as follows:

- Ensure that Kerberos authentication takes place successfully, by synchronizing the time on servers and workstations. The PDC Emulator in the forest root domain is considered the authoritative time source for the forest. All other Domain Controllers and Windows 2000 clients will automatically synchronize their time with the PDC Emulator. It is a good practice to synchronize the PDC Emulator to an Internet Network Time Protocol (NTP) time source. To do this, use the following command:

```
net time /setsntp: <NTP host name>
```

- Install the Directory Services Client on Windows 95/98 machines to ensure the use of NTLMv2 authentication, provide site awareness, and allow clients to search Active Directory.
- Ensure that Windows NT 4.0 clients have the latest service pack installed to ensure that NTLM version 2 is being utilized and NTFS version 5 has been installed.
- Manage the shares on multiple servers by creating a custom MMC with multiple copies of the Shared Folder snap-in tool, each with the focus on a different server that you are administering.
- Carefully plan your share file structure before implementing. As you plan, keep in mind how you can arrange the shares to make assigning and controlling permissions as simple as possible.
- Strongly resist changing permissions without good reasons after you have implemented the file structure. Ad hoc assignment of permissions will very quickly become unmanageable on a large network.
- Choose the most efficient way to assign permissions and work with inheritance by assigning the most restrictive permissions higher in the directory structure, and increasing the permissions further down the directory. For example, you may want to assign Read permissions to an upper-level directory, and then Change or Full Control on subfolders.
- Avoid using both NTFS and share permissions, which creates administrative complexity. In most cases, it is appropriate to leave the share permissions as Full Control for Everyone, and then set the permissions using NTFS. Using both methods of assigning permissions will become very complicated.
- Use the default permissions as much as possible. Keep things as simple as possible. The fact that you can get extremely detailed on file and folder permissions doesn't mean that you should.
- To ensure maximum data protection for all users' personal data, have them encrypt their home directory on the server.
- The Cipher command line tool can be used to set and view encryption settings. To view the options, open a command prompt and type Cipher /?.
- Be sure to audit Logon Failure to discover possible password hacking.
- Increase the default size of the event logs to handle higher amounts of entries.
- Use the Domain Controller Security Policy to audit Domain Controllers.
- Use the Domain Security Policy to set auditing on workstations and member servers.

## CHAPTER SUMMARY

- Kerberos authentication is the default authentication protocol used in Windows 2000 networks. If down-level clients, such as Windows NT, 95, or 98, are deployed throughout the network, LAN Manager or NTLM authentication will be used. The Directory Services Client will allow all down-level clients extra functionality, such as NTLM version 2 authentication, Active Directory site awareness, and searching capabilities in Active Directory. Certificate-based authentication is also supported in Windows 2000 to provide flexibility and security among multiple network platforms.
- When a user logs on, an access token is created that lists the user's personal Security Identifier (SID), and the SIDs of any groups that the user belongs to.
- All resources, such as folders, files, or printers, have an Access Control List (ACL), which contains a list of groups and users called Access Control Entries (ACEs) who have been assigned permissions to the resource.
- NTFS permissions are local permissions that can be assigned to a folder or file. When a user and a group that the user belongs to are both assigned permissions to a resource, cumulative permissions are applied to the user.
- Share permissions are folders that are made available over the network. When a user and a group that the user belongs to are both assigned permissions to a folder share, cumulative permissions are applied to the user.
- When both NTFS and share permissions are assigned to a resource, the most restrictive permissions are applied.
- The Encrypting File System allows a user to encrypt personal files to prevent access from unauthorized viewers. Only a designated recovery agent will be able to decrypt the file in the event of an emergency.
- A Windows 2000 Audit Policy should be designed to help make sure that the security goals are being met. Various categories can be audited that will list events such as logon successes or failures and file or directory access. Care must be taken not to audit too many categories, as unnecessary events may fill the event logs and make it difficult to manage effectively.

## KEY TERMS

**Access Control Entry (ACE)** — A single entry in an access control list that grants or denies permission to an NTFS or Active Directory object.

**Access Control List (ACL)** — A series of access control entries that define the level of access security that principals have to an NTFS or Active Directory object.

**access token** — Created for a user when the user logs on to a Windows 2000 computer or a Windows 2000 network. The access token includes the user's SID, the SIDs of all the groups to which the user belongs, and the user's rights and privileges.

**Authentication Service** — The service in Windows 2000 that gives users access to the network.

**auditing** — The option in Windows 2000 for monitoring administrative actions, logons and object access.

**Certification Authority (CA)** — A server that is used to grant and manage user and computer certificates in a Public Key Infrastructure.

**Discretionary Access Control List (DACL)** — Lists the security principals that have been assigned permission to the object, as well as the level of permissions for each security principal.

**Encrypted File System (EFS)** — The option available in Windows 2000 that enables a user to encrypt specified files on the computer hard disk. The encrypted files can be recovered only by the user and the designated recovery agent.

**Kerberos version 5** — The default authentication protocol used when Windows 2000 clients log on to a Windows 2000 domain.

**Key Distribution Center (KDC)** — The authenticating server in a Kerberos authentication model. In a Windows 2000 domain, the Domain Controllers are KDCs.

**NTFS permissions** — Used to control the level of access security that principals have to folders and files on an NTFS partition. NTFS permissions are applied both when a user logs on locally to a computer and when the user accesses the information through a network share.

**NTLM authentication (Windows NT Lan Manager)** — The authentication protocol used by down-level clients such as Windows NT when authenticating a user on a Windows 2000 network.

**recovery agent** — A user that has been assigned a recovery certificate so that the user can decrypt any file that has been encrypted by users. By default, the administrator on a standalone computer and the administrator of the first Domain Controller in a domain are the only recovery agents.

**Security Identifier (SID)** — A unique number identifying all security principals on a Windows 2000 network. Windows 2000 uses SIDs when managing permissions, rather than user or group account names.

**security principal** — Any object in Windows 2000 that can be used to assign permissions to other objects. Only users, groups, computers, and network services can be security principals.

**session ticket** — A ticket used in a Kerberos authentication model to gain access to computers and resources on a network.

**share permissions** — Permissions used to manage user access to shares on a Windows 2000 server. Share permissions are effective only when the user accesses the shares across a network connection, not when the user logs on the computer locally.

**System Access Control List (SACL)** — Lists the security principals whose access to a resource needs to be audited.

**Ticket Granting Ticket (TGT)** — A ticket granted to a user when the user logs on to a Windows 2000 domain. The TGT is used to acquire session tickets.



## REVIEW QUESTIONS

**3**

1. The first step in the Kerberos version 5 authentication process is:
  - a. the server request for the client certificate
  - b. the client request for a server certificate
  - c. the client request for a Ticket Granting Ticket
  - d. the client request for a Session Ticket
2. Before a Kerberos version 5 client with a valid Ticket Granting Ticket (TGT) can access a resource on a Windows 2000 server in the domain:
  - a. It must be authenticated by the Windows 2000 server.
  - b. It must apply to an Authentication Server for authentication.
  - c. It must apply to a Kerberos Key Distribution Center (KDC) for a session ticket.
  - d. It must check the Access Control List (ACL).
3. A Discretionary Access Control List is used to:
  - a. grant access to files
  - b. grant access to folders
  - c. grant access to printers
  - d. deny access to files
  - e. all of the above
4. If a user is given full control permissions to a folder, the user will also have full control permissions to:
  - a. printers in that folder
  - b. files in that folder
  - c. sub-folders in that folder
  - d. files and sub-folders in that folder
  - e. none of the above
5. Which file system is required to implement EFS?
  - a. FAT
  - b. FAT32
  - c. NTFS 5.0
  - d. NTFS 4.0
  - e. all of the above

6. Which security component allows you to track access to resources?
  - a. security groups
  - b. authentication
  - c. auditing
  - d. file encryption
  - e. credential management
7. What is the authentication protocol used when a Windows 2000 client is authenticating to a Windows 2000 Domain Controller?
  - a. NTLM
  - b. certificate-based authentication
  - c. smart cards
  - d. Kerberos version 5
  - e. clear text passwords
8. What level of permissions do you need to be able to create new shared printers?
  - a. Print
  - b. Share printers
  - c. Manage documents
  - d. Manage printers
9. An encrypted time stamp is included in Kerberos requests for access to network resources to:
  - a. Protect against a replay attack.
  - b. Be sure the time is accurately recorded in the log file.
  - c. Synchronize clocks between the client and the server.
  - d. Decrypt the session key.
10. In order to make sure that all shared file resources are secure, you have decided not to allow users to share information on their workstations. Where should you tell the users to place files that they want to share with other users?
  - a. an application folder on an application server
  - b. public shared folder on a file and print server
  - c. the user's home directory on a file and print server
  - d. the user's home directory on a RAS server

11. Why would you include the encrypting file system in your security plan?
  - a. to protect data in the case of hardware theft
  - b. to protect against attacks from the Internet
  - c. to protect against network packet sniffers
  - d. to protect against password guessing logon attacks
  - e. to protect against man-in-the-middle attacks
12. Which snap-in is required for MMC to view event logs?
  - a. Active Directory Users and Computers
  - b. performance logs and alerts
  - c. computer management
  - d. security templates
  - e. system information
13. Which log records auditing events?
  - a. application
  - b. DNS
  - c. file replication service
  - d. security
  - e. system
14. You have enabled the auditing of object access in Active Directory. What is the next step that you must perform if you wish to audit whenever anyone accesses a particular file?
  - a. Add the group Everyone to the audit entries for the file, and audit the success of List Folder/Read Data.
  - b. Add the group Domain Users to the audit entries for the file, and audit the success of Read Permissions.
  - c. Add the group Everyone to the audit entries for the file, and audit the success and failure of Read Permissions.
  - d. Add the group Domain Users to the audit entries for the file, and audit the failure of List Folder/Read Data.
15. You have limited space on the hard disk where you are storing the security logs and you want to make sure that the security logs do not fill up the remaining space. However, you do want to ensure that the most recent auditing information is always recorded. How should you configure the security log properties?
  - a. Overwrite events as needed.
  - b. Overwrite events older than X days.
  - c. Accept the default configuration.
  - d. Use a script to delete the logs every day.

16. Which of the following is not a security principal in a Windows 2000 domain?
  - a. a user object
  - b. an Organizational Unit
  - c. a security group object
  - d. a computer object
17. A Security Identifier (SID) must be:
  - a. unique to every security principal on a network
  - b. changed frequently
  - c. easily readable for all users
  - d. the same for all members of a group
18. Which advanced NTFS permission would you grant to allow a group to modify the contents of a file?
  - a. Create Files/Write Data
  - b. Create Folders/Append Data
  - c. Write Attributes
  - d. Write Extended Attributes
  - e. Change Permissions
19. By default, NTFS permissions are inherited from a folder to files and sub-folders inside that folder. How can you prevent this inheritance of permissions?
  - a. Block permission inheritance on the folder where the permissions were originally applied.
  - b. You cannot change the default permissions.
  - c. Block permission inheritance on the sub-folder.
  - d. Turn off the default configuration for the entire server.
20. You have just created a new share on a Windows 2000 server. What permissions are set on the share by default?
  - a. Deny Full Control to Domain Users
  - b. Allow Full Control to Domain Admins
  - c. Allow Full Control to Everyone
  - d. No permissions are in place by default
21. The default EFS recovery agent for a Windows 2000 Professional computer that is a part of a domain is the:
  - a. administrator of the local machine
  - b. administrator of the domain
  - c. Domain Admins group
  - d. No user is designated.

22. When a file is encrypted using EFS, the \_\_\_\_\_ is also encrypted and stored along with the encrypted data.
- a. the private key of the user
  - b. the private key of the recovery agent
  - c. the public key of the user
  - d. the public key of the recovery agent
  - e. the symmetrical encryption key of the file
23. What type of authentication is used with smart cards?
- a. Kerberos version 5
  - b. certificate-based authentication
  - c. digest authentication
  - d. Windows NT LAN Manager (NTLM)
  - e. RADIUS
24. Which type of authentication is used by older Windows operating systems such as Windows NT or Windows 95 (with the Directory Services Client installed) that are authenticating to a Windows 2000 Domain Controller?
- a. Kerberos version 5
  - b. certificate-based authentication
  - c. digest authentication
  - d. Windows NT LAN Manager (NTLM)
  - e. RADIUS
25. A user object has been Denied Read permission to a file, but the user is part of a security group that has been given Full Control of the file. When the user tries to access the file:
- a. He will have full control of the file.
  - b. He will be able to change the file, but not read it.
  - c. He will not be able to open the file.
  - d. He will be able to read, but not change the file.

## SETUP FOR HANDS-ON PROJECTS

The hands-on projects should meet the hardware requirements listed below:

Hardware Component	Windows 2000 Advanced Server
<b>CPU</b>	Pentium II 200 or higher
<b>Memory</b>	128 MB RAM
<b>Disk Space</b>	1 GB minimum for partition containing system files
<b>Drives</b>	CD-ROM Floppy Disk
<b>Networking</b>	TCP/IP 2 Network adapters Card 1 – 131.107.1.1: Label: Internal Card 2 – 131.107.2.1: Label: External Install DHCP but do not activate the scope (scope: 131.107.1.5 – 131.107.1.10)

1. Install Windows 2000 Advanced server. Name the computer **Server1**.
2. Run **DCPROMO** to upgrade the server to a Domain Controller. Install DNS when prompted. Use **Lonestar.com** as the domain name. Change the zone type to **Standard Primary**.
3. For the Domain Users group, add the right to log on locally to the Domain Controllers security policy.

## HANDS-ON PROJECTS



### Project 3-1

In this hands-on project, you will create a shared folder and edit the Access Control List (ACL) to only allow the Domain Administrators access to the share.

To create the shared folder:

1. Log on to your Windows 2000 computer as an administrator.
2. Right-click **My Computer** and click **Explore**.
3. In the Explorer window, click **Local Disk (C :)** within the left Folders pane.
4. Click the **File** menu, point to **New**, and click **Folder**.
5. Type **Data** as the name of the folder and press **Enter**.
6. Right-click the **Data** folder and click **Sharing**.
7. On the **Sharing** tab, click **Share this folder**. Leave the share name as Data.

8. To edit the Access Control List, click the **Permissions** button.
9. Click the **Everyone** group and click the **Remove** button.
10. Click the **Add** button.
11. Double-click the **Domain Admins** group. Click **OK**.
12. Click the **Allow** check box next to the **Full Control** permission. Click **OK** twice to return to the Explorer.
13. Close Explorer.



## Project 3-2

In this hands-on project, you will add a new user to the domain and compare how share level permissions are applied over the network connection versus at the local machine.

To create a new user:

1. Open Active Directory Users and Computers from the **Administrative Tools** menu.
2. Expand **Lonestar.com** and select the **Users** container.
3. Right-click the **Users** container and click **New, User**.
4. Add a user named **Bill Johnson** with the login name and password of **bill**.
5. Close all windows and log off as the administrator.
6. To test share level permissions, log on as Bill.
7. At the run command type **\\server1**.
8. Double-click the data share. Does Bill have access to the share? Why or why not?
9. Close the Server1 window.
10. Right-click **My Computer** and click **Explore**.
11. In the Explorer window, click **Local Disk (C :)** within the left Folders pane.
12. Double-click the **Data** folder in the right pane. Does Bill have access to the folder? Why or why not?
13. Close the Explorer window and log off.



## Project 3-3

In this hands-on project, you will increase access security to the data folder by applying NTFS permissions. You will then log on as Bill and test the new security settings.

To apply NTFS permissions:

1. Log on to your Windows 2000 computer as an administrator.
2. Right-click **My Computer** and click **Explore**.
3. In the Explorer window, click **Local Disk (C:)** within the left Folders pane.

4. Right-click the **Data** folder, and choose **Properties**.
5. Click the **Security** tab. Notice that the **Everyone** group has Full Control permissions. If the Everyone group has Full Control, why could Bill not access the folder over the network in the previous project?
6. Click the **Sharing** tab. Add the **Everyone** group with Full Access permissions.
7. Click the **Security** tab. Remove the inherited permissions by deselecting the check box next to **Allow inheritable permissions from parent to propagate to this object**.
8. On the **Security** warning, click **Remove**.
9. Click **Add** and select the **Domain Users** group. Click **OK**. Leave the default permissions.
10. Click **Add** and select **Domain Admins**. Click **OK**.
11. Click **Full Control** permissions for the **Domain Admins** group.
12. Click **OK** to and close **Windows Explorer**.
13. At the run command type `\\server1`.
14. Double-click the **data** share. Does the administrator have access to the share? Why or why not?
15. Create a new text document called **doc1.txt**. To complete this step, right-click in the data window and choose **New, Text Document**.
16. Close all windows and log off.
17. To compare share level permissions to NTFS permissions, log on as Bill.
18. At the run command, type `\\server1`.
19. Double-click the **Data** share. Does Bill have access to the share? Why or why not?
20. Right-click in the data window and choose **New, Text Document**. Is Bill able to create a new document? Why or why not?
21. Close all windows and log off.



## Project 3-4

In this hands-on project, you will monitor shared folder availability and access using the computer management console.

To view and manage shared folders:

1. Log on to your Windows 2000 computer as an administrator.
2. At the run command, type `\\server1`.
3. Double-click the **Data** share. Minimize all windows.
4. Right-click **My Computer** and click **Manage**.
5. In the left pane, expand the **Shared Folders** node.



6. Click the **Shares** sub-node. Notice all of the shares available on the local computer.
7. Select the **Sessions** and the **Open Files** sub-nodes. Notice that the administrator is connected to the Data folder.
8. Create a new shared folder called **Reports**. Make sure that Administrators have full access and that other users have read only permissions. To perform this step, right-click the **Shares** sub-node and select **New File Share**. To create the new folder, click the **Browse** button.
9. Close all windows and log off.



## Project 3-5

In this hands-on activity, you will verify that only an EFS recovery agent is able to read the contents of an encrypted file.

1. Log on to your Windows 2000 computer as an administrator.
2. Open Active Directory Users and Computers and create two new user accounts, **Bob Jones** and **Jeff Smith**.
3. Close all windows and log off.
4. To create an encrypted file, log on as **Bob Jones**.
5. Right-click **My Computer** and click **Explore**.
6. Create a new folder called **Bobs files** on the C drive.
7. Right-click the **Bobs files** folder and click **Properties**.
8. In the **Properties** dialog box, click the **Advanced** button at the bottom of the General tab.
9. Click the check box next to **Encrypt contents to secure data**. Click **OK** twice.
10. Create a new text file named **ENCRYPT.TXT** in the Bobs files folder.
11. Close all windows and log off.
12. To attempt to access an encrypted file, log on as **Jeff Smith**.
13. Right-click **My Computer** and click **Explore**.
14. Attempt to view the contents of the file in the **Bobs files** folder. Are you able to view the folder contents?
15. Close all windows and log off.
16. To access an encrypted file as a recovery agent, log on as the **Administrator**.
17. Click **Start**, point to **Programs**, point to **Administrative Tools**, and select **Domain Security Policy**.
18. Expand the **Security Settings** node in the left pane.
19. Expand the **Public Key Policies** sub-node.

20. Click **Encrypted Data Recovery Agents**. Note that Administrator is the default recovery agent. Close the **Domain Security Policy** console.
21. Open **Windows Explorer** and attempt to view the contents of the file in the **Bobs Files** folder. Are you able to open the ENCRYPT.TXT file? Why or why not?
22. Close all windows and log off.



## Project 3-6

In this hands-on project, you will enable auditing for all failed log ons and any access to the Notepad application. Any time a domain user fails to log on or when users run NOTEPAD.EXE, events will be logged to the appropriate event log. *Hint:* Look for events 560 and 529.

To enable auditing:

1. Log on to your Windows 2000 computer as an administrator.
2. Click **Start**, point to **Programs**, point to **Administrative Tools**, and click **Domain Controller Security Policy**.
3. Expand the **Security Settings** node in the left pane.
4. Expand the **Local Policies** node.
5. Click the **Audit Policy** node. The details pane lists the various auditing configuration selections.
6. Double-click **Audit logon events**.
7. Select the check box to define this policy, and choose to audit the failed log ons. Click **OK**.
8. Double-click **Audit object access**.
9. Select the check box to define this policy, and choose to audit all successes. Click **OK**. Close the Domain Controller security policy console.
10. Right-click **My Computer** and click **Explore**.
11. Browse to **C:\Winnt\System32**. Click **Show Files** if necessary to view the details pane.
12. Right-click **NOTEPAD.EXE** and click **Properties**.
13. Click the **Security** tab and then click the **Advanced** button.
14. Click the **Auditing** tab and then click the **Add** button.
15. Double-click the **Domain Users** group.
16. Enable the **Successful** checkbox for **Read Permissions**. Click **OK** three times.
17. Close all windows and log off.



## Project 3-7

In this hands-on project, you will attempt to log on with an invalid password and then successfully log on and run the Notepad application. You will then view the results of the audit.

To test the audit policies:

1. At the Log on prompt, press **Ctrl+Alt+Delete**. Type your name as the user name and **1234** as the password.
2. Click **OK** at the logon message. Repeat Steps 1 and 2 two more times.
3. Log on as the **Administrator**.
4. Click **Start**, point to **Programs**, point to **Accessories**, and open the **Notepad** application.
5. Close **Notepad**.
6. Click **Start**, point to **Programs**, point to **Administrative Tools**, and click **Event Viewer**.
7. Click **Security Log**.
8. Double-click the various events to view the details. What information does the audit provide?
9. Close all windows and log off.

## CASE PROJECTS



### Case Project 3-1

The management at Southdale Property Management is concerned about the level of file security within the office. Some of the financial information that is stored on the file servers is quite confidential and should be accessible only by the managers of the company. The most important documents on the file server are the financial documents, which include the current financial information on all the buildings owned by the company, the amount of money invested by each investor, and salary information about each employee. The managers should be able to view this information, but only the CFO and the accountant should be able to modify the data. As well, the CEO wants to be able to track all changes to this information, including information on who made the changes and when they were made. Other information should be accessible to all employees, although each employee should also have a location on the file server that is accessible only by the employee.

Three of the employees are responsible for investigating new buildings that the company may be interested in buying. These employees also negotiate any purchases. As a result, they spend a significant amount of their time outside the office, and have been issued

laptop computers. These employees have very confidential information on their laptops, including contract information. The management at Southdale Property Management is concerned about the security of this information.

1. As part of your security plan, you must provide security for the files at the company office. What would be the key components of your security planning to provide this security?
2. How could you ensure the security of the data on the laptop computers?



### Case Project 3-2

As a financial institution, Fleetwood Credit Union has very high security needs. There are a number of security-related issues that have been raised recently that the company would like to resolve as quickly as possible.

1. Currently almost all the desktop computers at head office are running Windows 98. The management team would like a recommendation on whether upgrading the computers to Windows 2000 will result in a significant security improvement. The company does not like to spend money unless there is a very clear business benefit.
2. There is no clear policy to indicate where users will store their files and the type of security that is configured on the files. This problem came to the attention of the management team when a manager told her assistant to use her computer while the assistant's computer was being fixed, and the assistant inadvertently deleted a whole folder of confidential information on the manager's computer. The management team wants to make sure that this never happens again by restricting access to all folders and by making sure that all data is backed up every night. What would you recommend?
3. There is a problem with printer security. This came to light when one of the financial advisors printed a confidential document on a public printer, and then got distracted by a phone call before he could pick up the document. When he finished the phone call, he went to pick up the document, but it was gone. No one knows who took the document, or even if it printed properly. How can you make sure that this will never happen again?